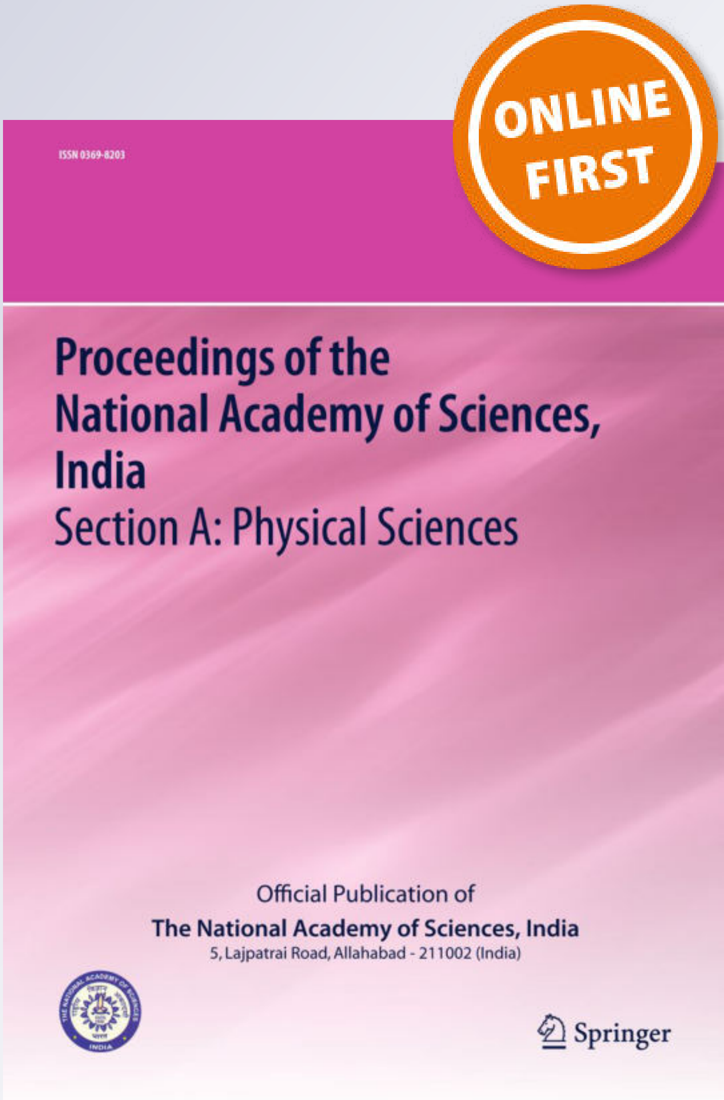
FRBPSO: A Fuzzy Rule Based Binary PSO for Feature Selection

Shikha Agarwal, R. Rajesh & Prabhat Ranjan

Proceedings of the National Academy of Sciences, India Section A: Physical Sciences

ISSN 0369-8203

Proc. Natl. Acad. Sci., India, Sect. A Phys. Sci.
DOI 10.1007/s40010-017-0347-8



Springer

Proc. Natl. Acad. Sci., India, Sect. A Phys. Sci.
DOI 10.1007/s40010-017-0347-8

RESEARCH ARTICLE

FRBPSO: A Fuzzy Rule Based Binary PSO for Feature Selection

Shikha Agarwal[1] · R. Rajesh[2] · Prabhat Ranjan[1]

Received: 26 December 2015 / Revised: 26 July 2016 / Accepted: 30 January 2017

Abstract Particle swarm optimization and fuzzy logic have shown their fruits for many years across the fields of science. Fuzzy logic acts as an intelligent layer to any conventional system. Recently fuzzy logic has been used to improve the performance of particle swarm optimization (PSO). This paper presents a novel fuzzy rule based binary PSO (FRBPSO) for feature selection to get better classification and a survey on the PSO fuzzy logic hybrid methods. The results on benchmarking high dimensional microarray datasets show the merits of the proposed FRBPSO method.

Keywords Particle swarm optimization · Fuzzy logic · Fuzzy rule based PSO · Classification · Feature selection

1 Introduction

High dimensionality is a well-known challenge in which numbers of features are very high when compared to the numbers of samples [1]. Dimension reduction is the common approach to deal with challenges of dimensionality. Many statistical and computational methods have been reported in literature [2–7] for dimensionality reduction. These methods can be grouped into two categories; feature selection and feature extraction.

Particle swarm optimization (PSO) [8] is one of these computational approaches for feature selection which has shown its merits in many fields of research due to its cognitive/social behavior, exploitation/exploration capability and faster convergence [9]. Basic PSO is a population based optimization algorithm designed for real valued space. Kennedy and Eberhart in 1997 developed Binary PSO (BPSO) [10] for the discrete binary variables.

Despite of many advantages, PSO has some drawbacks of getting into local optimum and stagnation. To overcome these problems, many variants of PSO have been proposed by many researchers. Fuzzy PSO is one of the variants of PSO in which fuzzy logic's strength of uncertainty handling is incorporated into PSO to make it more suitable for the optimum result for different applications.

This paper presents a survey on the PSO fuzzy logic hybrids for the last one decade, which reveals that in most of the fuzzy PSO variants only parameters of PSO has been optimized using fuzzy logic. Therefore, in this paper a novel fuzzy rule based binary PSO (FRBPSO) has been proposed in which uncertainty in feature selection is handled using fuzzy logic. The results on benchmarking dataset show the merits of proposed FRBPSO.

2 Particle Swarm Optimization

Particle swarm optimization (PSO) works based on the sharing/learning of information from the past and mimics the searching for food by a flock of birds [8].

In PSO a swarm is made up of some particles (candidate solutions). Each particle (each bird in the flock) represents

✉ Shikha Agarwal
shikhaagarwal@cub.ac.in; agarwal.shikha.ibm@gmail.com

R. Rajesh
kollamrajeshr@ieee.org

Prabhat Ranjan
prabhatranjan@cub.ac.in

1 Department of Computer Science, Central University of South Bihar, Patna 800014, India

2 Department of Computer Science, Central University of Kerala, Kasaragod, India

a point in the search space and it moves towards the optimum solution through learning and sharing from its own experience and from the experience of other neighbouring particles. Each particle is associated with two components position and velocity. Position component represents a specific solution and velocity represents the direction of movement of particle.

The PSO algorithm starts with the random initialization of position X (where $X_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$ represents the position of ith particle) and velocity V (where $V_i = [v_{i1}, v_{i2}, \ldots, v_{im}]$ represents the velocity of ith particle) of n particles (each of dimension m). In next step fitness of particles is determined using fitness function. There is no single fitness function for evaluating the candidate particle. Fitness evaluation function varies from problem to problem.

For example, in classification problem one of the most commonly used criterion function is the classification accuracy obtained by candidate particle.

In case of data clustering problem Emami et al. [11] has used Eq. (1) as objective function for fitness.

$$F = \sum_{j=1}^{K} \sum_{i=1}^{N} u_{ij}^{m} d_{ij} \tag{1}$$

The goal of above Eq. (1) is to cluster N data points into K clusters based on degree of membership (u_{ij}^{m}) of ith data point in cluster j. d_{ij} denotes the distance of ith data point from jth cluster center.

In case of fuzzy controller design problem Wong et al. [12] have proposed a variable fitness function using rise time and integral absolute error corresponding to candidate solution.

In this paper classification accuracy obtained using K-nearest neighbour classifier is used as a fitness value of the corresponding feature subset solution.

In PSO position and velocity are updated using the following equations (Eqs. 2 and 3) proposed by Kennedy and Eberhart [8, 13].

$$v_{id}^{new} = v_{id}^{old} + vp_{id} + vg_{id} \tag{2}$$

$$x_{id}^{new} = x_{id}^{old} + v_{id}^{new} \tag{3}$$

where x_{id}^{old} and v_{id}^{old} are previous position and velocity of dth feature in the ith particle respectively. x_{id}^{new} and v_{id}^{new} are new position and velocity of dth feature in the ith particle respectively.

vp_{id} and vg_{id} are defined as

$$vp_{id} = c_1 r_1 (pb_{id} - x_{id}^{old}), \tag{4}$$

$$vg_{id} = c_2 r_2 (gb_d - x_{id}^{old}) \tag{5}$$

Personal best pb of ith particle is a vector of length m, defined as $pb_i = [pb_{i1}, pb_{i2}, \ldots, pb_{im}]$, where pb_{id} is the personal best feature value of the dth feature in the ith particle. Global best gb of population is a vector of length m, defined as $gb = [gb_1, gb_2, \ldots, gb_m]$, where gb_d is the global best feature value of the dth feature for all particles. c_1 & c_2 are constants, and r_1 & r_2 are random numbers. Later an inertia weight (ω) is multiplied along with v_{id}^{old} as shown in Eq. (6) to balance between exploration and exploitation [13].

$$v_{id}^{new} = \omega * v_{id}^{old} + vp_{id} + vg_{id} \tag{6}$$

In binary PSO (BPSO) [10] each particle is represented by binary bits (a vector of 0's and 1's) to solve discrete problem. In terms of feature selection '0' in binary particle represents the absence of the feature and '1' represents the presence of the feature corresponding to the position of the bit. In BPSO, Eq. (6) remains the same, but instead of Eq. (3), the following if-else statement is used-

$$if\ (rand() < S(v_{id}^{new}))\ then\ x_{id}^{new} = 1\ else\ x_{id}^{new} = 0$$

where rand() is a random number generation function and S is a sigmoid limiting function which is used to map the velocity value in the range of 0–1.

Improved binary PSO resets the gb_d if there is no change in the last three iterations in the dth feature's fitness to overcome the trapping into local optima [14]. Rajesh et al. [15] has proposed four modification (MIBPSO1-4) in IBPSO. MIBPSO-3 is the third modification in which random number of if-then-else statement of IBPSO is replaced with 0.5.

3 A Review of PSO with Fuzzy Logic

3.1 Fuzzy Discrete Particle Swarm Optimization [16, 17]

Pang et al. [16] developed a hybrid method with fuzzy logic and PSO to solve the discrete travel salesman problem. The aim is to get closed short length tour which visits each city only one time (Hamiltonian cycle).

Let $S = (S_1, S_2, \ldots, S_n)$ be a solution of a TSP, where n and S_i are the number of cities and the ith node in the tour respectively. Let $N = (N_1, N_2, \ldots, N_n)$ be the serial number of the cities. The degree of membership for assigning a city N_j to ith node S_i is given by $r_{ij} = \mu_R(S_i, N_j)$ such that $0 < r_{ij} < 1$, where R is the fuzzy relation and μ_R is the membership function. Position and velocity of particles for the PSO algorithms are defined as follows.

$$X = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{bmatrix} \tag{7}$$

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ v_{n1} & v_{n2} & \cdots & v_{nn} \end{bmatrix} \tag{8}$$

Initially position and velocity matrix are generated randomly with $\sum r_{ij} = 1$ and $\sum v_{ij} = 0$. The negative elements in the position matrix are converted to zero. Inorder to normalize the position matrix, the elements of each row are divided by the respective row total. Modified equations are suggested by the authors to update the position and velocity of each particle.

$$\begin{aligned} V_i(t+1) = w \otimes V_i(t) \oplus (c_1 r_1) \otimes (pb_i(t) \ominus X_i(t)) \\ \oplus (c_2 r_2) \otimes (gb_i(t) \ominus X_i(t)) \end{aligned} \tag{9}$$

$$X_i(t+1) = X_i(t) \oplus V_i(t+1) \tag{10}$$

$\omega \otimes V(t)$ means multiply the each element of velocity matrix with ω. $\oplus$ and $\ominus$ represents a simple matrix addition and subtraction respectively. For defuzzification of the position matrix the authors use *"Max number"* method. For each row they select the maximum of the values that are not selected by the previous rows and then select the corresponding column index as one of the element in the solution set S.

Fuzzy PSO of Pang et al. [16] was utilized by Izakian et al. [17] for fuzzy clustering. Here the data objects are $S = \{S_1, S_2, S_3, \ldots, S_n\}$ mapped into the set of cluster centers $N = \{N_1, N_2, N_3, \cdots, N_c\}$. The velocity and position updation are similar to that of the Eqs. (9) and (10) of Fuzzy PSO by Pang et al. [16]. $f(X) = K/J_m$ is the fitness function, where K and J_m are constant and objective function of fuzzy c-means algorithm. The algorithm of fuzzy PSO for clustering starts with the initialization of the position matrix, velocity, personal best (*pb*) and global best (*gb*). Then the following steps are repeated until the termination condition is met. (1) Calculate the cluster center using $N_j = \frac{\sum_{i=1}^{n} r_{ij} S_j}{\sum_{i=1}^{n} r_{ij}^{m}}$, (2) Calculate the fitness of each particle based on fitness function, (3) Find *pb* and *gb*, (iv) Update the velocity and position using Eqs. (9) and (10). Izakian et al. [17] also proposed a hybrid FCM-FPSO for having the advantages of both FCM and FPSO. In this hybrid algorithm FCM is called after FPSO and is repeated until termination condition is met.

Emami and Derakhshan have proposed clustering algorithms by hybridizing fuzzy k-mean clustering algorithm with PSO [11]. In FKMPSO basic principles of two algorithms (Fuzzy Logic and k-means) are kept same. Working of FKMPSO is as follows; first initialize the population which is a matrix of dimension N × (k × D), where N is number of candidate solution or population size, k is number of clusters and D is the dimension of data set. Then PSO search starts and fitness of each particle is calculated using fuzzy k mean clustering. At the end, global best is selected as optimum clustering solution. This FKMPSO algorithm helps the FKM to escape from local optima.

3.2 Charisma Based PSO [18]

Abdelbar et al. [18] proposed a generalized form of PSO with several particles in neighborhood influencing based on certain degree of *charisma* (a fuzzy variable). In Charisma PSO the velocity equation is given by

$$\begin{aligned} v_{id}^{new} = \omega v_{id}^{old} + (c_1 r_1)(pb_{id} - x_{id}^{old}) \\ + \sum_{h \in B(i,k)} (c_2 r_2 \psi(h))(p_h - x_{id}^{old}) \end{aligned} \tag{11}$$

where $B(i, k)$ denotes the k-best particles in the neighborhood of ith particle, p_h is the fitness of any one of the charismatic particle, and $\psi(h)$ is the charisma value of particle h (and is a Cauchy function) given by Eq. (12) with $f(p_g)$ as the best particle in the neighborhood & l as a user defined value. If k = 1 then Eq. (11) reduces to the standard PSO.

$$\psi(h) = \frac{1}{1 + \left(\frac{f(p_h) - f(p_g)}{\frac{f(p_g)}{l}} \right)^2} \tag{12}$$

3.3 Fuzzy Adaptive Turbulent PSO [19]

Fuzzy Adaptive Turbulent PSO [19] which works like a turbulent pump was proposed by Liu et al. to control the particles velocity [19]. The velocity of the particles are controlled by the following equation

$$v_{ij} = \begin{cases} v_{ij}, & \text{if } |v_{ij}| > \; = v_c \\ r v_{max}/\rho, & \text{if } |v_{ij}| < v_c \end{cases} \tag{13}$$

where r, ρ and v_c are respectively, the random number in the range of $[-1, 1]$, the scaling factor and the minimum velocity threshold. Inorder to control the parameters, namely, the ρ and v_c, a Mamdani type fuzzy controller is designed with two inputs and two outputs. The "normalized current best performance evaluation (NCBPE)" is used as one of the inputs with three Gaussian membership functions in the fuzzy logic controller and is given by $NCBPE = \frac{CBPE - CBPE_{min}}{CBPE_{max} - CBPE_{min}}$, where, $CBPE_{min}$ is actual minimum, $CBPE$ is the current best performance evaluation, $CBPE_{max}$ is bad performance considering the problem as a minimization problem. Current velocity is another input to the fuzzy controller with two trapezoidal membership functions. The $Vck = \frac{e - v_c}{10} - 1$ is one of the outputs of the fuzzy controller with three triangular membership functions. The other output variable is ρ with three triangular membership functions. Six rules are formed based on the input variables.

3.4 Fuzzy Adaptive PSO [20]

Inorder to adaptively change the inertia weight, Shi et al. [20] proposed a fuzzy adaptive inertia weight system. The input variables are "normalized current best performance evaluation (NCBPE)" and current inertia weight. The output of the fuzzy system is the change in inertia weight. The authors have used three membership functions for each of the input/output variables and formed nine fuzzy rules [20].

3.5 Fuzzy Adaptive Catfish PSO [21]

In Catfish PSO, Chuang et al. [22] introduced Catfish particle, in which the velocity of ten percentage of the particles at extreme points of the search space (max or min) are reset to zero. In Fuzzy Adaptive Catfish PSO [21], Chuang et al. [22] used fuzzy adaptive inertial weight system [20] with Catfish PSO.

3.6 Fuzzy Particle Swarm Optimization with Cross Mutation [23]

The fuzzy PSO with cross mutation (FPSOCM) was proposed by the Chai et al. in which separate fuzzy systems are used to adaptively predict the inertia weight and the control parameters of cross mutation [23]. FPSOCM is used to develop a neural network classifier for the three mental task problem where the weights of neural network are trained using FPSOCM.

Another extension is a Type-2 fuzzy adaptive binary particle swarm optimization with single mutation operator proposed by Soeprijanto et al. in which Type-2 FIS has been used for tuning of the parameters (inertia weight ω and learning factors c_1 and c_2) of BPSO [24].

3.7 Multiobjective PSO

Torabi et al. proposed a multi objective PSO (MOPSO) for parallel machine scheduling problem and makes use of fuzzy selection methods for guide selection methods (guides are particles that are used instead of gbest) [25]. Ganguly et al. extended MOPSO and proposed heuristic selection of guides MOPSO (HSG-MOPSO) based on fuzzy-Pareto-dominance for the electrical distribution system [26]. Another interesting work is by Dinh et al. where hybrid MOPSO with simulated annealing is used for the optimization of the linguistic variable's parameters and fuzzy rule selection for the classification problem [27].

3.8 Dynamic Parameter Adaptation Through Fuzzy Logic [28]

Olivas et al. have incorporated new improvement in the divergence and convergence of PSO using fuzzy logic inference system. Iteration (ratio of the current number of iteration to the maximum number of iterations), diversity (defined as average of euclidean distance between each particle and global best particle) and error (average of difference between fitness of each particle and global best particle) are used as input to the fuzzy inference system. For each input, three triangular membership functions are used. c_1 and c_2 of Eqs. (4) and (5) are used as output parameters of fuzzy inference system with five triangular membership function for each.

Some of the interesting PSO fuzzy hybrids are, PSO with a fuzzy controller to adaptively change the inertia weight and learning coefficient based on increment in global best (IGO) and deviation of particle fitness value (DEV) [29], fuzzy PSO in which inertia weight and social/cognitive learning factors are changed for each particle using fuzzy controller [30] and, fuzzy PSO with cross mutation in which inertia weight and parameter of cross mutation are adaptively changed using fuzzy logic [31].

Other extensions of PSO fuzzy hybrids are, quantum fuzzy PSO (QPSO) [32], ring topology based binary PSO [33], wrapper approach of BPSO with neural networks, fuzzy models and support vector machines [34], Hybrid PSO with fuzzy reasoning [35], hybrid Nelder–Mead fuzzy adaptive PSO [36], enhanced comprehensive learning cooperatively PSO with fuzzy inertia weight [37], kernelized rough set fuzzy c-mean PSO clustering [38], attractive repulsive fully informed PSO using self organizing population mechanism [39] and, ANFIS PSO [40].

Some of the interesting applications of fuzzy logic with PSO include, but not limited to, fuzzy min–max neural network PSO for intrusion detection using fuzzy hyber box [41], fuzzy logic based adaptive PSO for optimum covering array generation for software testing problem [42], PSO fuzzy based controller to optimize the multi area power network [43], location identification for chaff points using PSO in fuzzy vault based biometric crypto system [44], optimizing the voltage of energy conservation system with suitably weighting the objective function using fuzzy logic and PSO [45], fuzzy radial basis function neural network PSO for PID control system [46], Microarray gene expression data clustering using fuzzy logic and PSO [47], type-2 fuzzy c-partitioning using PSO for image segmentation [48], power and voltage control using fuzzy adaptive PSO for distributed network [49], fuzzy logic based PSO for gas leakage detection system [50], weighted fuzzy interpolation reasoning method with dynamic weight updation using PSO [51], fuzzy PSO simplex method for designing of PM coupling [52], longitudinal controller for intelligent vehicle design using fuzzy logic and PSO [53], fuzzy PSO collaborative filtering for recommender system [54], on-line tuning of controller feedback filter using fuzzy logic PSO [55], interval type-2 Takagi Sugeno fuzzy system optimization using PSO and SVM [56].

4 Fuzzy Rule Based Binary Particle Swarm Optimization (FRBPSO)

Fuzzy logic helps to handle uncertainty, vagueness and ambiguity in a much better way than any traditional logic in many applications. Fuzzy logic systems will also behave like an extra layer of intelligence to any existing systems. Hence there is no negative consequence of incorporation of fuzzy logic in PSO. Moreover, inclusion of fuzzy logic in any conventional model brings more flexibility, reduces the development time, makes the computation easier, combines the logic reasoning with power of mathematics. Fuzzy logic could fine tune any traditional computational concept to fit into non linear ambiguous situation due to its approximation capability [57].

The literature survey clearly reveals that, fuzzy logic is incorporated in PSO for the purpose of parameter optimization to enhance the strength of PSO. Previously proposed models are not utilizing the ability of fuzzy logic to handle vagueness, ambiguity and uncertainty to improve the optimization capacity of PSO. Therefore, in this section, a novel fuzzy rule based binary PSO has been proposed to handle the uncertainty of feature selection problem.

4.1 Problem in PSO and Fuzzy PSO

In most of the variants of PSO, the difference between current position & previous personal best position [see Eq. (4)] and the difference between current position & global best position [see Eq. (5)] are considered to be in unit time step. Moreover, position differences are added to the velocity component [see Eq. (2)] and this addition of position to velocity does not justify the law of physics.

The three parameters (ω, c_1, c_2) and many random numbers in the basic PSO, make the model highly uncertain and vulnerable to deviation from global optimum. Moreover, assigning appropriate value to each of these parameters is itself a optimization problem.

4.2 Proposed Method FRBPSO

To reduce the dependency on equations and to handle the vagueness and uncertainty of feature selection incorporated by parameters and random numbers, fuzzy rule based binary PSO (FRBPSO) has been proposed.

Reconsider the Eq. (6) of PSO with three terms, first one representing the previous velocity (memory of velocity), second one representing a factor of the difference between the personal best (*pb*) position & the current position (cognitive component) and, the third one representing a factor of the difference between the global best (*gb*) position & the current position (social component). By carefully observing the equation, the notion of adding a velocity with differences of position needs some rethinking.

In this work in place of unit time step, the time difference between the time of obtaining the previous personal best (iteration number of the previous personal best) and the current time (current iteration number) has been calculated which is named as $p_{timestep}$. Similarly, the time difference between the time of obtaining the previous global best and the current time is calculated which is named as $g_{timestep}$. Now, the velocity of personal best (vp_{id}) and the velocity of global best (vg_{id}) is given by Eqs. (14) and (15)

$$vp_{id} = ((c_1 r_1 (pb_{id} - x_{id}^{old}))/p_{timestep}) \tag{14}$$

$$vg_{id} = ((c_2 r_2 (gb_{id} - x_{id}^{old}))/g_{timestep}) \tag{15}$$

The components of PSO like memory of velocity (v_{id}^{old}), new cognitive component [vp_{id} Eq. (14)] and new social component [vg_{id} Eq. (15)] are needed to be best utilized to run the algorithm. In our proposed FRBPSO all the three components are taken in consideration as input to fuzzy system to modify position update strategy of binary PSO.

In order to construct the fuzzy rules, vp_{id}, vg_{id} and v_{id}^{old} are converted into fuzzy input variables. Ranges of vp_{id}, vg_{id} and v_{id}^{old} are $[-2\ 2], [-2\ 2]$ and $[-6\ 6]$ respectively. The fuzzy linguistic terms of the inputs, namely, vp_{id}, vg_{id} and v_{id}^{old} are shown in Fig. 1. Each input is granulated into two Gaussian fuzzy membership functions. Output variable x_{id}^{new} takes singleton values namely, H and L. H takes 0.8 (accept the feature) and L takes 0.2 (reject the feature).

Eight fuzzy rules are formed to predict the new position as shown in the Table 1. When social and personal learning are high, then feature at that position is accepted and when social and personal learning both are low then feature at that position is rejected. When vp_{id} and vg_{id} are getting different linguistic values (Social and personal learning are opposite to each other) then the position is decided based on memory input (v_{id}^{old}). Final output binary feature vector is obtained by converting the output of FIS to the nearest integer (0 or 1).

Figure 2 shows 3-D plot of possible values of x corresponding to the values of vg_{id} and v_{id}^{old}. Figure 3 shows 3-D plot of possible values of x corresponding to the values of vp_{id} and v_{id}^{old}. Figure 4 shows 3-D plot of possible values of x corresponding to the values of vp_{id} and vg_{id}.

4.3 Fitness and Performance Metrics

Fitness of each particle is obtained using K-nearest neighbor classifier to obtain personal best and global best position in the swarm. To apply KNN on binary string of

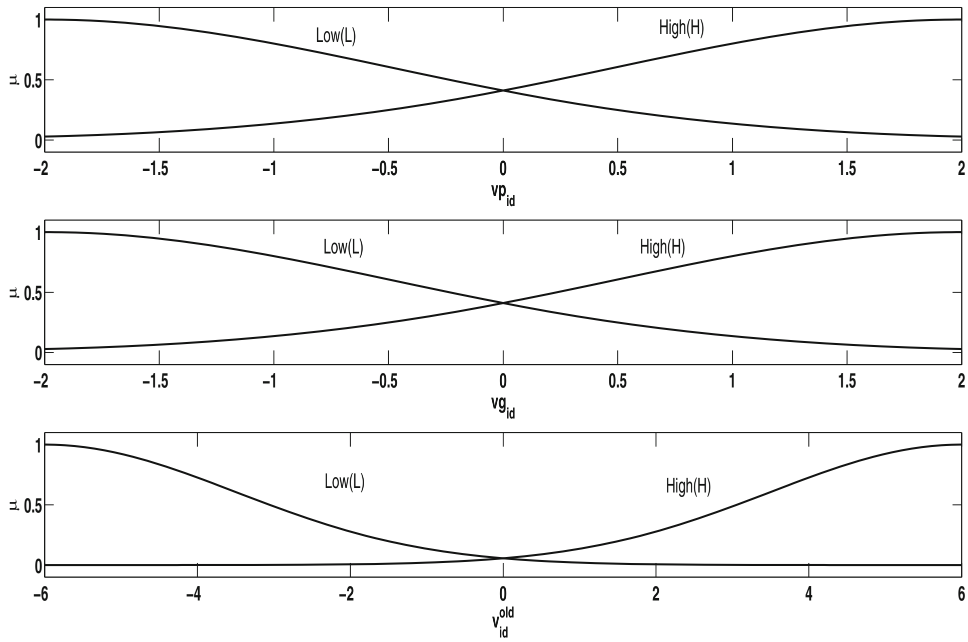


Fig. 1 Fuzzy inputs vp_{id}, vg_{id} and v_{id}^{old}. Each input is having two Gaussian fuzzy membership functions [Low(L) and High(H)]. The variance and mean of vp_{id} and vg_{id} are set to [1.5 −2] and [−1.5 2] for L and H membership functions respectively. For v_{id}^{old} variance and mean for L and H member functions are [−2.5 −6] and [2.5 6] respectively

Table 1 Fuzzy rules for FRBPSO

If v_{old} = L and vp_{pd} = H and vg_d = H	Then x = H
If v_{old} = H and vp_{pd} = L and vg_d = H	Then x = H
If v_{old} = L and vp_{pd} = L and vg_d = H	Then x = L
If v_{old} = H and vp_{pd} = H and vg_d = L	Then x = H
If v_{old} = L and vp_{pd} = H and vg_d = L	Then x = L
If v_{old} = H and vp_{pd} = L and vg_d = H	Then x = H
If v_{old} = H and vp_{pd} = L and vg_d = L	Then x = L
If v_{old} = L and vp_{pd} = L and vg_d = L	Then x = L

particle, dataset features which are selected in candidate solution are taken into consideration and data is classified using those features only.

For example in a particle of size 10 with position vector ‘0011011101’ only 3rd, 4th, 6th, 7th, 8th and 10th features are selected from all samples of the data set, then data is classified using KNN based on these selected features. Classification accuracy of each particle is treated as a fitness value of the particle.

The performance of algorithms is computed according to classification accuracy and selected features metrics. The fitness of global best candidate solution at the end of the searching is considered as the classification accuracy of the algorithm and the number of features selected in the global best solution is considered as final selected features by the algorithm.

The goal of this work is to find a particle with minimum number of features and maximum classification accuracy. At the end candidate solution with most benefit(classification accuracy) with least cost (number of features) is selected as optimal feature subset as solution of the feature selection problem.

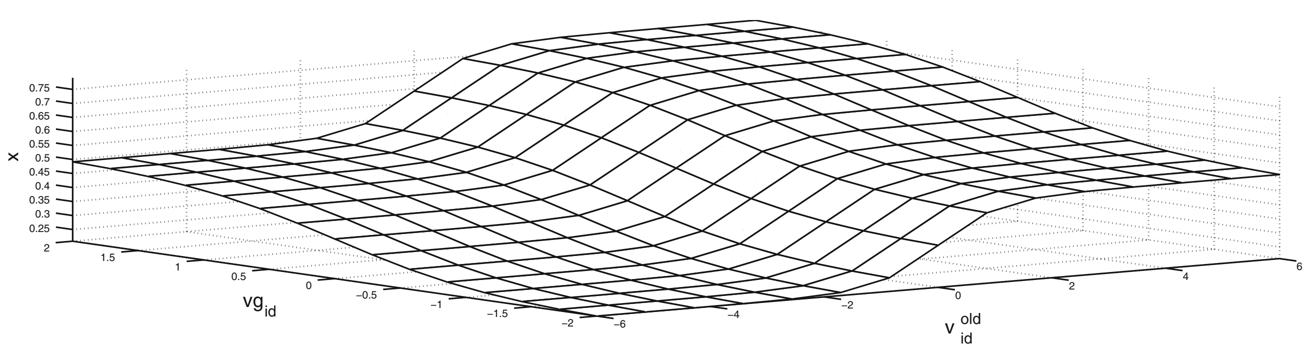


Fig. 2 3-D plot of possible values of x corresponding to the values of vg_{id} and v_{id}^{old} provided by the FRBPSO fuzzy rules (see Table 1)

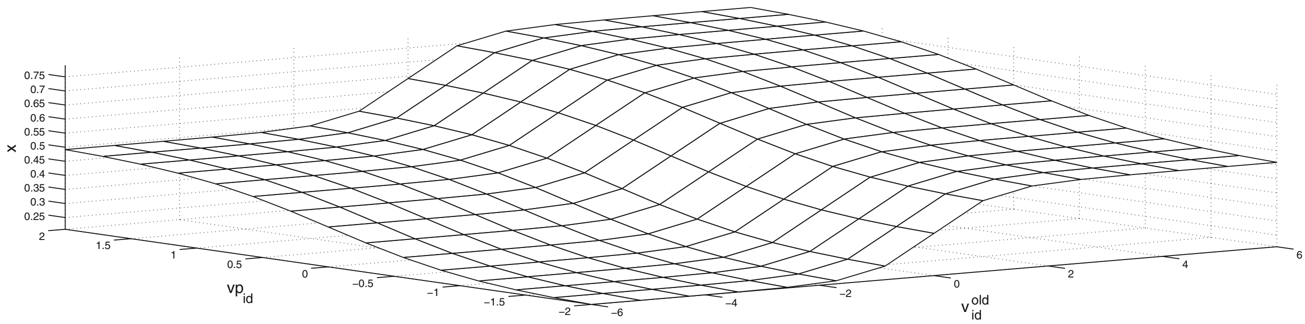


Fig. 3 3-D plot of possible values of x corresponding to the values of vp_{id} and v_{id}^{old} provided by the FRBPSO fuzzy rules (see Table 1)

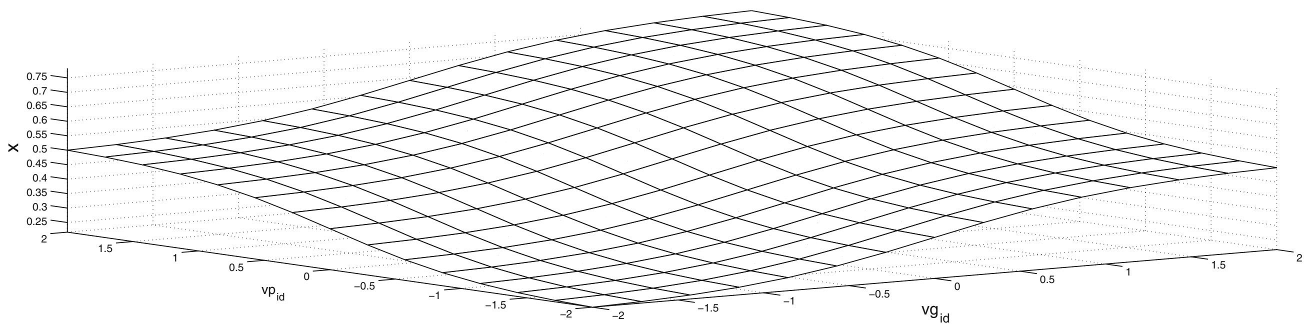


Fig. 4 3-D plot of possible values of x corresponding to the values of vp_{id} and vg_{id} provided by the FRBPSO fuzzy rules (see Table 1)

4.4 Scope and Application

Proposed FRBPSO method is a feature selection algorithm which could be apply to any data in which dimensionality is very high. For example NMR spectroscopic data, text document data, chemo-informatics data, stock market data and handwriting data etc.

5 Experimental Setup

In order to show the performance of FRBPSO, seven benchmarking gene expression profile microarray data sets [58–61] are used. The details of each data set is given in Table 2.

The quality of selected subset of features are evaluated based on the classification accuracy of K- nearest neighbor (KNN) classifier using leave one out cross validation (LOOCV) method. The value of k for KNN classifier is set to 1, learning rates of both c_1 and c_2 are set to 2, velocity of particles are set to be limited in the range of $[-6,\ 6]$. The number of particles in swarm is set to 40 and algorithms are allowed to iterate for 150 iterations with stopping criteria set to $max\_iteration = 150$ or $max\_accuracy = 100$.

In this experiment proposed method is compared with the KNN (classifier with all features), IBPSO, MIBPSO-3. IBPSO is considered as a most competitive method, since Chuang et al. claims in their work that IBPSO method outperforms all the other binary PSOs [14]. Reported results are average values over ten runs of each method on each data set.

Table 2 Description of Gene expression profile data sets

Dataset name	No. of samples	No. of features	No. of classes
SRBCT	83	2308	4
DLBCL	77	5469	2
Brain_Tumor1	90	5920	5
Brain_Tumor2	50	10,367	4
Leukemia1	72	5327	3
Leukemia2	72	11,225	3
Prostate_Tumor	102	10,509	2

6 Results and Discussions

Table 3 shows the average classification accuracy, the average number of selected features and standard deviation of the accuracy obtained from KNN, IBPSO, MIBPSO-3 and FRBPSO. From Table 3 it is clearly visible that FRBPSO is out performing the other methods in terms of average classification accuracy and selected features.

Figure 5 shows the bar graph of classification accuracy obtained from KNN, IBPSO, MIBPSO-3 and FRBPSO.

Table 3 Average classification accuracy, selected features and standard deviation of accuracy using KNN, IBPSO, MIBPSO and FRBPSO

Data set	Method	Classification accuracy	Selected features	Standard deviation of accuracy
SRBCT	KNN	91.57	ALL	
	IBPSO-KNN	97.59	1124	0.5040
	MIBPSO-KNN	98.05	275	0.0162
	FRBPSO	98.19	213	0.0117
DLBCL	KNN	87.01	ALL	
	IBPSO-KNN	94.81	2697	$2.3e^{-16}$
	MIBPSO-KNN	96.16	282	0.0012
	FRBPSO	96.49	105	0.0174
Brain1	KNN	86.67	ALL	
	IBPSO-KNN	89.99	2924	0.4802
	MIBPSO-KNN	89.88	1160	0.0222
	FRBPSO	90.67	803	0.0141
Brain2	KNN	70.00	ALL	
	IBPSO-KNN	81.2	4983	0.3936
	MIBPSO-KNN	85.4	1207	0.0358
	FRBPSO	87.6	662	0.0343
Leukemia1	KNN	87.50	ALL	
	IBPSO-KNN	97.59	2643	0.4949
	MIBPSO-KNN	98.05	987	0.00620
	FRBPSO	98.89	825	0.0144
Leukemia2	KNN	70.00	ALL	
	IBPSO-KNN	97.22	4958	0.5197
	MIBPSO-KNN	87.22	2257	0.0068
	FRBPSO	97.50	1028	0.0128
Prostate	KNN	76.47	ALL	
	IBPSO-KNN	91.14	1029	0.646
	MIBPSO-KNN	90.39	426	0.0321
	FRBPSO	92.43	418	0.0178

Figure 6 shows the bar graph of number of selected features obtained from IBPSO, MIBPSO-3 and FRBPSO. In Fig. 6 KNN is not shown because it uses all features which are very large in numbers for each dataset, hence it is eliminated from Fig. 6.

Figures 5 and 6 also show a considerable increase in classification accuracies and graceful decrease in number of features selected using FRBPSO in all datasets.

Table 3, Figs. 5 and 6 reveal the blessings of dimensionality reduction with increased classification accuracy after feature selection as compared to without feature selection.

Obtained classification accuracy versus number of iterations for SRBCT, DLBCL, Brain1, Brain2, Leukemia1, Leukemia2 and Prostate Tumor using FRBPSO are shown in Figs. 7, 8, 9, 10, 11, 12 and 13 respectively. Overlapping lines in the plots show, the result of multiple runs of FRBPSO on same data set. From Figs. 7, 8, 9, 10, 11, 12 and 13, it is clearly revealed that multiple runs of FRBPSO are converging in the same manner (highly overlapping), which confirm the consistency and stability of the FRBPSO algorithm.

Out performing experimental results of FRBPSO is justifying the advantages of using fuzzy inference system (FIS) in PSO search. Use of fuzzy logic in updating the position of particle helps to handle uncertainty of trapping in local optimum solution to an extent.

Inclusion of time factor in velocity components (Eqs. 14 and 15) helps FRBPSO to get satisfactory results when compared to other methods of feature selection.

Fuzzy incorporation in PSO resulted in robust model in which global optima is obtained in shorter development time with simple decision process as compared to other traditional PSO methods.

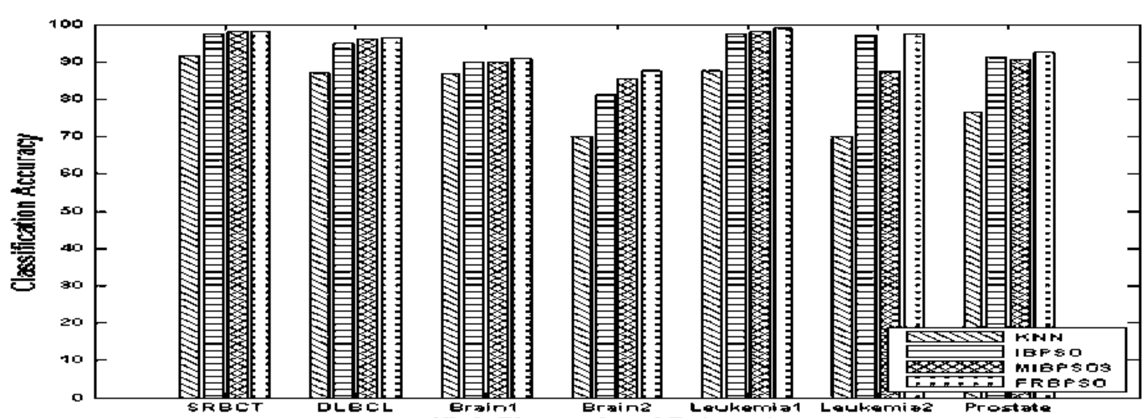


Fig. 5 Average classification performance of KNN, IBPSO, MIBPSO3 and FRBPSO

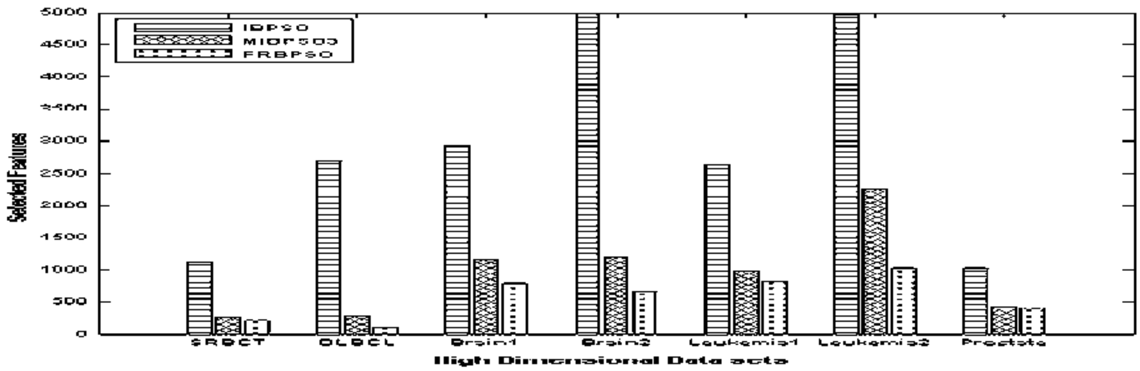


Fig. 6 Average number of selected features using IBPSO, MIBPSO3 and FRBPSO

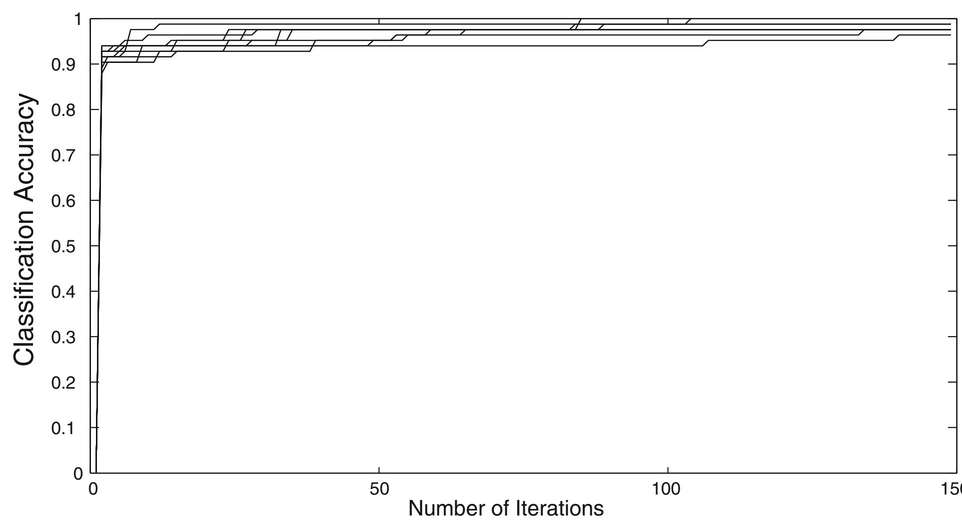


Fig. 7 Performance of FRBPSO on SRBCT dataset

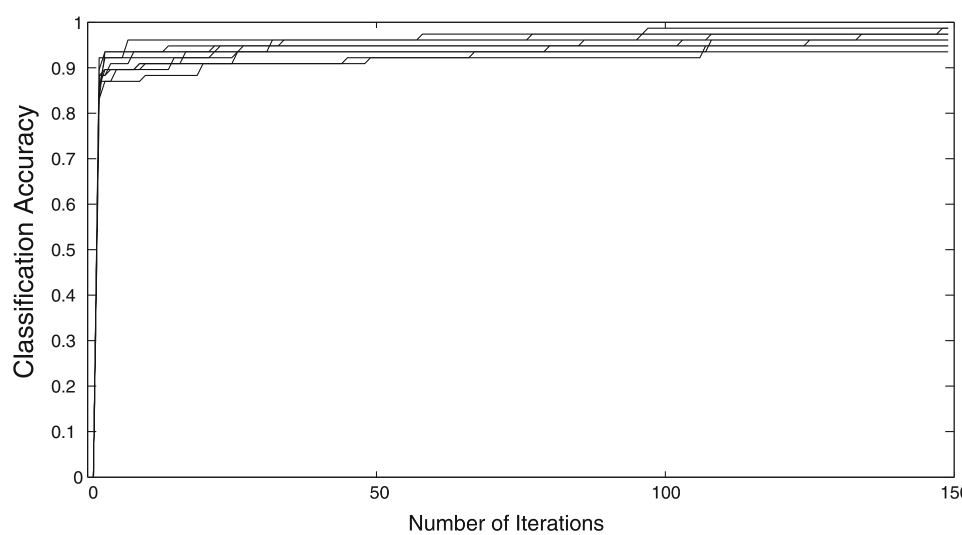


Fig. 8 Performance of FRBPSO on DLBCL dataset

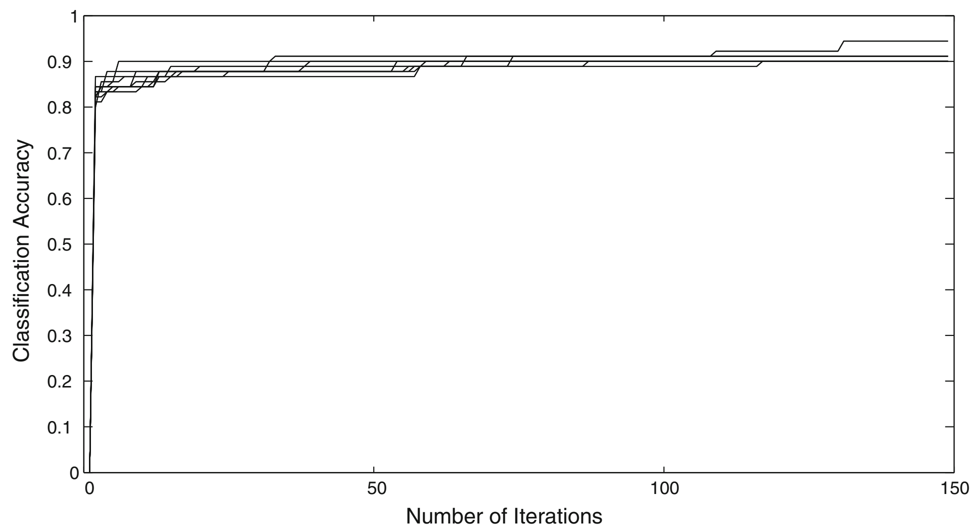


Fig. 9 Performance of FRBPSO on Brain1 dataset

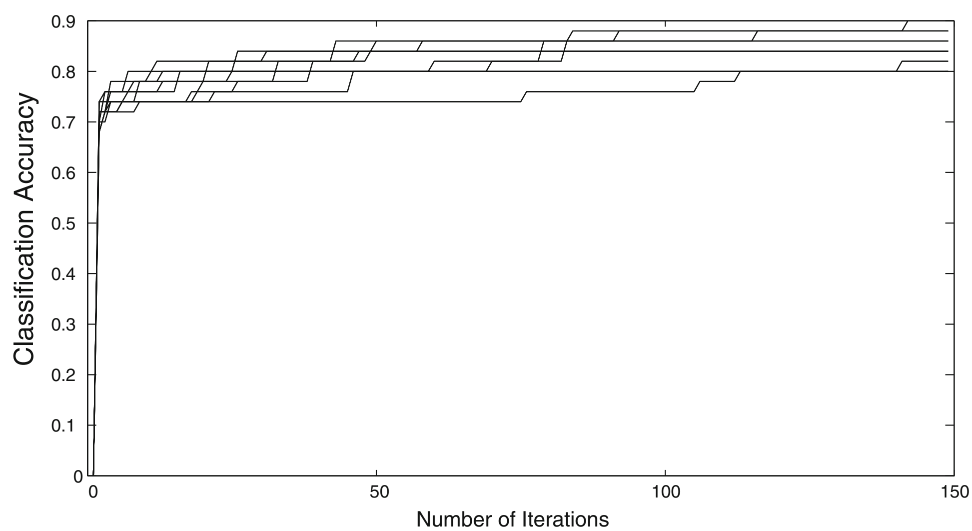


Fig. 10 Performance of FRBPSO on Brain2 dataset

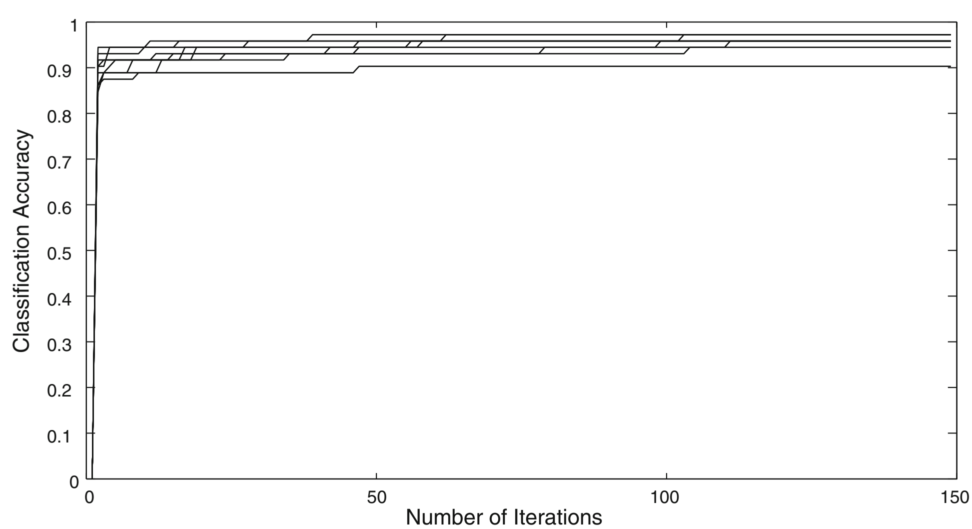


Fig. 11 Performance of FRBPSO on Leukemia1 dataset

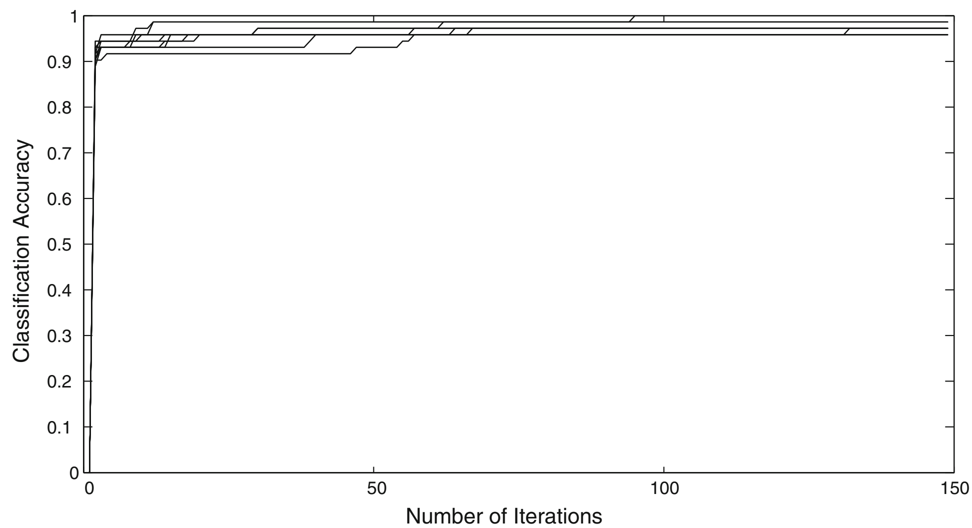


Fig. 12 Performance of FRBPSO on Leukemia2 dataset

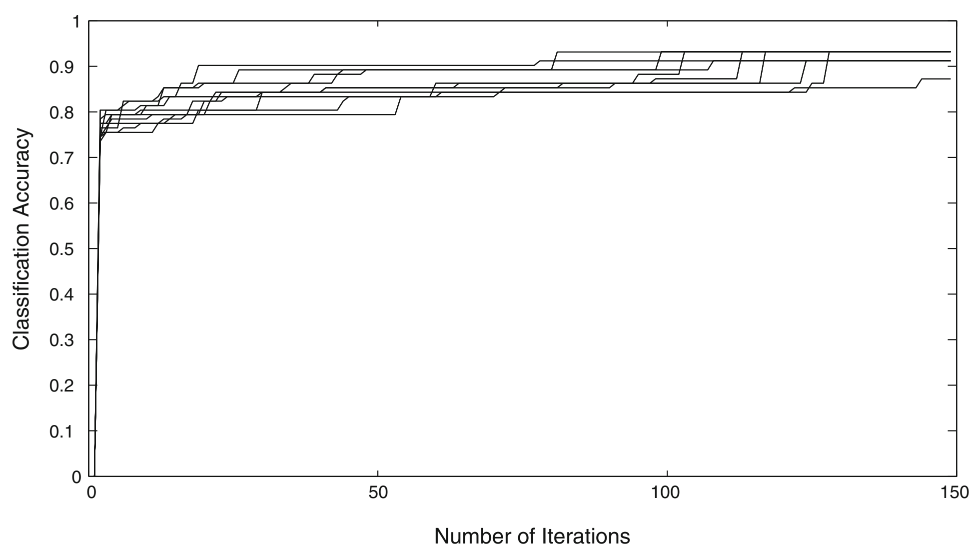


Fig. 13 Performance of FRBPSO on Prostate Tumor dataset

7 Conclusion

Particle swarm optimization has seen its fruits in many applications. In this paper a survey of fuzzy logic hybrid with PSO is carried out which concludes that in most of the cases of fuzzy logic is used for PSO parameter optimization. Hence, in this paper a novel fuzzy rule based binary PSO (FRBPSO) for feature selection to achieve promising classification accuracy is proposed in which fuzzy rules are designed to find the best position of the particle (Selected Features). Experimental results on benchmarking gene expression profiles show the merits of FRBPSO. The proposed FRBPSO could be used across the fields of high dimensional data to find the global optimum with satisfactory consistency and convergence.

Acknowledgements Shikha Agarwal, awarded with Senior Research Fellowship (Grant No. (09/1144(0001)2015EMR-I) wish to acknowledge the financial support of Council of Scientific & Industrial Research (CSIR), India.

References

1. Donoho DL (2000) High-dimensional data analysis: the curse and blessings of dimensionality. In: American Mathematical society conference Math challenges of the 21st Century. American Mathematical Society. http://www-stat.stanford.edu/~donoho/Lectures/CBMS/Curses.pdf
2. Wedderburn RWM (1974) Quasi-likelihood functions, generalized linear models, and the Gauss–Newton method. Biometrika 61:439–477
3. Fan J, Lv J (2008) Sure independent screening for ultra-high dimensional feature space. J R Stat Soc B 70:849–911
4. Hansen LP (1982) Large sample properties of generalized method of moments estimators. Econometrica 50:1029–1054
5. Yuan Y-X (1999) Step size of gradient method AMS/IP studies in advance mathematics. American Mathematical Society, Providence 42:785

6. Nesterov Y (2012) Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM J Optim 22:341–362
7. Efron B, Hastie T, Johnstone I, Tibshirani R (2004) Least angle regression. Ann Stat 32:407–499
8. Kennedy J, Eberhart RC (1995) Particle swarm optimization. IEEE international conference on neural networks. Perth, Australia, pp 1942–1948
9. Bergh F, Engelbrecht AP (2004) A cooperative approach to particle swarm optimization. IEEE Trans Evol Comput 8:225–239
10. Kennedy J, Eberhart RC (1997) A discrete binary version of the particle swarm algorithm. In: IEEE conference on systems, man, and cyber, pp 4104–4108
11. Emami H, Derakhshan F (2015) Integrating fuzzy K-means, particle swarm optimization, and imperialist competitive algorithm for data clustering. Arab J Sci Eng 40:3545–3554
12. Wong CC, Wang HY, Li SA (2008) PSO-based motion fuzzy controller design for mobile robot. Int J Fuzzy Syst 10(1):284–292
13. Kennedy J, Eberhart RC, Shi Y (2001) Swarm intelligence. Morgan Kaufman, San Mateo
14. Chuang LY, Chang HW, Tu CJ, Yang CH (2008) Improved binary PSO for feature selection using gene expression data. Comput Biol Chem 32:29–38
15. Rajesh R, Agarwal S (2014) Some modification in particle swarm optimization. The 18th online world conference on soft computing in industrial applications
16. Pang W, Wang KP, Zhou CG, Dong LJ (2004) Fuzzy discrete particle swarm optimization for the solving travelling salesman problem. In: Proceedings of fourth international conference on computer and information technology, pp 796–800
17. Izakian H, Abraham A, Snasel V (2009) Fuzzy clustering using hybrid fuzzy c means and fuzzy particle swarm optimization. In: Proceedings of the world congress on nature & biologically inspired computing (NaBIC), pp 1690–1694
18. Abdelbar AM, Abdelbar S, Wunsch DC (2005) Fuzzy PSO: a generalization of particle swarm optimization. In: Proceedings of international joint conference on neural networks, Montrael, Canada, pp 1086–1091
19. Liu H, Abraham A, Zhang W (2007) Fuzzy adaptive turbulent particle swarm optimization. Int J Innov Comput Appl (IJICA) 1:39–47
20. Shi Y H, Eberhart R C (2001) Fuzzy adaptive particle swarm optimization. In: Proceedings of the IEEE congress on evolutionary computation, pp 101–106
21. Chuang LY, Tsai SW, Yang CH (2012) Fuzzy adaptive catfish particle swarm optimization. Artif Intell Res 1:149–170
22. Chuang LY, Tsai SW, Yang CH (2008) Catfish particle swarm optimization. In: Proceedings of IEEE swarm intelligence symposium, St. Louis, Missouri, pp 1–5
23. Chai R, Ling SH, Hunter GP, Tran Y, Nguyen HT (2014) Brain computer interface classifier for wheelchair commands using neural network With fuzzy particle swarm optimization. IEEE J Biomed Health Inf 18:1614–1624
24. Soeprijanto A, Abdillah M (2011) Type 2 fuzzy adaptive binary particle swarm optimization for optimal placement and sizing of distributed generation. In: Proceedings of the 2nd international conference on instrumentation, communications, information technology, and biomedical engineering (ICICIBME), IEEE, pp 233–238
25. Torabi SA, Sahebjamnia N, Mansouri SA, Bajestani MA (2013) A particle swarm optimization for a fuzzy multi-objective unrelated parallel machine scheduling problem. Appl Soft Comput 13:4750–4762
26. Ganguly S, Sahoo NC, Das D (2013) Multi-objective particle swarm optimization based on fuzzy-Pareto-dominance for possibilistic planning of electrical distribution system incorporating distributed generation. Fuzzy Set Syst 213:47–73
27. Dinh P, Thanh TN, Xuan TT (2014) A Hybrid Multi-objective PSO-SA algorithm for the fuzzy rule based classifier design problem with the order based semantics of linguistic terms. VNU Journal of Science: Comp. Sci Com Eng 30:44–56
28. Olivas F, Valdez F, Castillo O (2015) Fuzzy classification system design using PSO with dynamic parameter adaptation through fuzzy logic. Fuzzy Log Augment Nat Inspir Optim Metaheuristics Stud Comput Intell 574:29–47
29. Tian DP, Li NQ (2009) Fuzzy particle swarm optimization algorithm. International joint conference on artificial intelligence, IEEE, pp 263–267
30. Aminian E, Teshnehlab M (2013) A novel fuzzy particle swarm optimization. In: Proceedings of the 13th Iranian conference on fuzzy systems (IFSC), IEEE, pp 1–6
31. Ling SH, Chan KY, Leung FHF, Jiang F, Nguyen H (2016) Quality and robustness improvement for real world industrial systems using a fuzzy particle swarm optimization. Eng Appl Artif Intel 47:68–80
32. Zhong Q, Yao M, Jiang W (2010) Quantum fuzzy particle swarm optimization algorithm for image clustering. International conference on image analysis and signal processing (IASP), IEEE, pp 276–279
33. Moaref A, Naeini VS (2013) A particle swarm optimization based on a ring topology for fuzzy rough feature selection. 13th Iranian conference on fuzzy systems (IFSC), IEEE, pp 1–6
34. Vieira SM, Mendonca LF, Farinha GJ, Sousa JMC (2012) Metaheuristics for feature selection: Application to sepsis outcome prediction. IEEE congress on evolutionary computation (CEC), IEEE, pp 1–8
35. Li NJ, Wanga WJ, Hsu CCJ (2015) Hybrid particle swarm optimization incorporating fuzzy reasoning and weighted particle. Neurocomputing. 167:488–501
36. Nesamalar JJD, Venkatesh P, Raja SC (2016) Managing multiline power congestion by using Hybrid Nelder–Mead–Fuzzy adaptive particle swarm optimization (HNMFAPSO). Appl Soft Comput 43:222–234
37. Gholamian M, Meybodi MR (2015) Enhanced comprehensive learning cooperative particle swarm optimization with fuzzy inertia weight (ECLCFPSOIW). J comput Robot 8:57–66
38. Halder A (2015) Kernel based rough fuzzy c Means clustering optimized using particle swarm optimization. International symposium on advanced computing and communication (ISACC), pp 41–48
39. Mo S, Zeng J, Xu W (2016) Attractive and repulsive fully informed particle swarm optimization based on the modified fitness model. Soft Comput Fusion Found Methodol Appl 20:863–884
40. Rini DP, Shamsuddin SM, Yuhaniz SS (2016) Particle swarm optimization for ANFIS interpretability and accuracy. Soft Comput Fusion Found Methodol Appl 20:251–262
41. Azad C, Jha VK (2016) Fuzzy min–max neural network and particle swarm optimization based intrusion detection system. Microsyst Technol. Springer, Berlin
42. Mahmoud T, Ahmed BS (2015) An efficient strategy for covering array construction with fuzzy logic based adaptive swarm optimization for software testing use. Expert Syst Appl 42(22):8753–8765
43. Dhillon SS, Lather JS, Marwaha S (2015) Multi area load frequency control using particle swarm optimization and fuzzy rules. In: Procedia computer science, 3rd international conference on recent trends in computing, vol 57, pp 460–472
44. Amirthalingam G, Radhamani G (2016) New chaff point based fuzzy vault for multimodal biometric cryptosystem using particle swarm optimization. J King Saud Univ Comput Inf Sci 28:381–394
45. Manbachi M, Farhangi H, Palizban A, Arzanpour S (2016) Smart grid adaptive energy conservation and optimization engine

utilizing particle swarm optimization and fuzzification. Appl Energy 174:69–79
46. Wang Y, Chenxie Y, Tan J, Wang C (2015) Fuzzy radial basis function neural network PID control system for a quadrotor UAV based on particle swarm optimization. In: Information and automation, IEEE international conference, pp 2580–2585
47. Das R, Saha S (2015) Gene expression classification using a fuzzy point symmetry based PSO clustering technique. Second international conference on soft computing and machine intelligence (ISCMI), pp 69–73
48. Ouarda A (2015) Image thresholding using type 2 fuzzy c partition entropy and particle swarm optimization algorithm. Computer vision and image analysis applications (ICCVIA), international conference on, pp 1–7
49. Chen S, Hu W, Su C, Zhang X (2015) Optimal reactive power and voltage control in distribution networks with distributed generators by fuzzy adaptive hybrid particle swarm optimization method. IET Gener Transm Dis 9(1096):1103
50. Miraswan KJ, Maulidevi NU (2016) Particle Swarm Optimization and fuzzy logic control in gas leakage detector mobile robot. In: International conference on automation, cognitive science, optics, micro electromechanical system, and information technology (ICACOMIT), IEEE Explorer
51. Chen SM, Hsin WC (2015) Weighted fuzzy interpolative reasoning based on the slopes of fuzzy sets and particle swarm optimization techniques. IEEE Trans Cybern 45:1250–1261
52. Wakeel ASE, Smith AC (2015) Hybrid fuzzy particle swarm optimization simplex (FPSOS) algorithm for optimum design of PM drive couplings. Electr Power Compon Syst 43:1560–1571
53. Thanok S, Parnichkun M (2015) Longitudinal control of an intelligent vehicle using particle swarm optimization based sliding mode control. Adv Robot 29:525–543
54. Wasid M, Kant V (2015) A particle swarm approach to collaborative filtering based recommender systems through fuzzy features. Procedia Comput Sci 54:440–448
55. Hashim HA, Ferik SE, Abido MA (2015) A fuzzy logic feedback filter design tuned with PSO for L1 adaptive controller. Expert Syst Appl 42:9077–9085
56. Du Y, Lu X, Chen L, Zeng W (2016) An interval type 2 TS fuzzy classification system based on PSO and SVM for gender recognition. Multimed Tools Appl 75:987–1007
57. Albertos P, Sala A (1998) Fuzzy logic controllers. Advantages and drawbacks. In: VIII congreso latinoamericano de control automático, congreso de la asociación chilena de control automático, ANALES III, ACCA
58. Statnikov A, Tsamardinos I, Dosbayev Y, Aliferis CF (2005) GEMS: a system for automated cancer diagnosis and biomarker discovery from microarray gene expression data. Int J Med Inf 7:491–503
59. Statnikov A, Aliferis CF, Tsamardinos I, Hardin D, Levy S (2005) A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics 21:631–43
60. Statnikov A, Aliferis C F, Tsamardinos I (2015) Methods for multi-category cancer diagnosis from gene expression data: a comprehensive evaluation to inform decision support system development. Medinfo, p 813
61. Statnikov A, Aliferis C F, Tsamardinos I, www.gems-system.org, Discovery Systems Laboratory, Department of Biomedical Informatics, Vanderbilt University, Nashville